

**JP2000163246A**

**2000-6-16**

**Bibliographic Fields**

**Document Identity**

(19)【発行国】	(19) [Publication Office]
日本国特許庁(JP)	Japan Patent Office (JP)
(12)【公報種別】	(12) [Kind of Document]
公開特許公報(A)	Unexamined Patent Publication (A)
(11)【公開番号】	(11) [Publication Number of Unexamined Application]
特開2000-163246(P2000-163246A)	Japan Unexamined Patent Publication 2000-163246 (P2000-163246A)
(43)【公開日】	(43) [Publication Date of Unexamined Application]
平成12年6月16日(2000.6.16)	2000 June 16* (2000.6.16)

**Public Availability**

(43)【公開日】	(43) [Publication Date of Unexamined Application]
平成12年6月16日(2000.6.16)	2000 June 16* (2000.6.16)

**Technical**

(54)【発明の名称】	(54) [Title of Invention]
FIFO機能を有する記憶装置サブシステム	STORAGE DEVICE SUBSYSTEM WHICH POSSESSES FIFO FUNCTION
(51)【国際特許分類第7版】	(51) [International Patent Classification, 7th Edition]
G06F 5/06	G06F 5/06
3/06 301	3/06 301
12/00 514	12/00 514
13/38 310	13/38 310
340	340
【FI】	[FI]
G06F 5/06 B	G06F 5/06 B
3/06 301 M	3/06 301 M
12/00 514 A	12/00 514 A
13/38 310 B	13/38 310 B
340 C	340 C
【請求項の数】	[Number of Claims]
6	6
【出願形態】	[Form of Application]
OL	OL
【全頁数】	[Number of Pages in Document]
15	15

【テーマコード(参考)】

5B0655B0775B082

【F ターム(参考)】

5B065 BA01 CA07 CC08 CC10 CS06 CS10  
ZA20 5B077 AA41 NN07 5B082 FA02**Filing**

【審査請求】

未請求

【(21)【出願番号】】

特願平10-333726

【(22)【出願日】】

平成10年11月25日(1998. 11. 25)

**Parties****Applicants**

【(71)【出願人】】

【識別番号】

000005108

【氏名又は名称】

株式会社日立製作所

【住所又は居所】

東京都千代田区神田駿河台四丁目6番地

**Inventors**

【(72)【発明者】】

【氏名】

北村 学

【住所又は居所】

神奈川県川崎市麻生区王禅寺1099番地 株式会社日立製作所システム開発研究所内

【(72)【発明者】】

【氏名】

山神 嘉司

【住所又は居所】

神奈川県川崎市麻生区王禅寺1099番地 株式会社日立製作所システム開発研究所内

【(72)【発明者】】

【Theme Code (For Reference)】

5B0655B0775B082

【F Term (For Reference)】

5B065 BA 01 CA07 CC08 CC10 cs 06 CS1 0 ZA20 5B077  
AA41 NN07 5B082 FA02

【Request for Examination】

Unrequested

【(21) [Application Number]】

Japan Patent Application Hei 10- 333726

【(22) [Application Date]】

1998 November 25\* (1998.11.25)

【(71) [Applicant]】

【Identification Number】

000005108

【Name】

HITACHI LTD. (DB 69-054-1503)

【Address】

Tokyo Chiyoda-ku Kanda Surugadai 4-Chome 6

【(72) [Inventor]】

【Name】

Kitamura \*

【Address】

Kanagawa Prefecture Kawasaki City Asao-ku Ozenji  
1099address Hitachi Ltd. (DB 69-054-1503) system  
developmental research laboratory \*

【(72) [Inventor]】

【Name】

crest \* Kenji

【Address】

Kanagawa Prefecture Kawasaki City Asao-ku Ozenji  
1099address Hitachi Ltd. (DB 69-054-1503) system  
developmental research laboratory \*

【(72) [Inventor]】

## 【氏名】

荒川 敬史

## 【住所又は居所】

神奈川県川崎市麻生区王禅寺1099番地 株式会社日立製作所システム開発研究所内

## (72)【発明者】

## 【氏名】

本間 繁雄

## 【住所又は居所】

神奈川県小田原市国府津2880番地 株式会社日立製作所ストレージシステム事業部内

**Agents**

## (74)【代理人】

## 【識別番号】

100096954

## 【弁理士】

## 【氏名又は名称】

矢島 保夫

**Abstract**

## (57)【要約】

(修正有)

## 【課題】

システム間のデータ転送において、ディスク装置等の記憶装置を介したデータ転送を行うこと、および複数のインターフェースを備えるディスク装置に対し、先入れ先出し機構を備える。

## 【解決手段】

記憶装置サブシステム3は2つのインターフェース31、32を有し、ディスク33は各ホストからのシーケンシャルアクセスのみが許可される。

すなわちホスト1が一旦ディスク33にデータを書き込むとホスト2から当該アドレスのデータを読み出さない限りホスト1から当該アドレスにデータを再書き込みできないように制御する。

これにより、ディスク33がFIFOとして機能する。

## [Name]

Arakawa Takashi

## [Address]

Kanagawa Prefecture Kawasaki City Asao-ku Ozenji  
1099address Hitachi Ltd. (DB 69-054-1503) system  
developmental research laboratory \*

## (72) [Inventor]

## [Name]

\*\* Shigeo

## [Address]

Kanagawa Prefecture Odawara City Kouzu 2880address  
Hitachi Ltd. (DB 69-054-1503) storage unit systems  
department \*

## (74) [Attorney(s) Representing All Applicants]

## [Identification Number]

100096954

## [Patent Attorney]

## [Name]

Yajima Yasuo

## (57) [Abstract]

(There is an amendment.)

## [Problems to be Solved by the Invention]

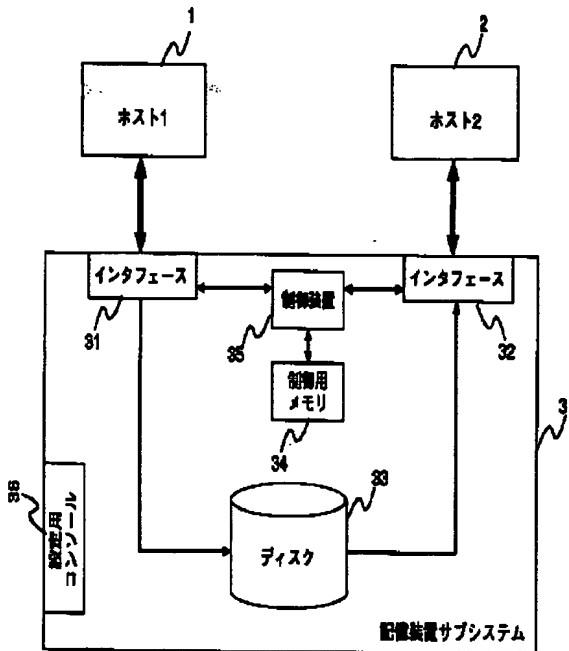
In data transfer between system , is through disk drive or other storage device data transfer isdone, it has first-in, first-out mechanism and vis-a-vis disk drive which has interface of plural .

## [Means to Solve the Problems]

storage device subsystem 3 has 2 interface 31 , 32, disc 33 is done only sequential access from each host grant .

Namely when host 1 writes data to disc 33 once, if data of this said address is not read out from host 2, from host 1 in the this said address data rewriting in order for it not to be possible, itcontrols.

Because of this, disc 33 it functions as FIFO .



## Claims

### 【特許請求の範囲】

#### 【請求項 1】

第1の計算機および第2の計算機とそれぞれ接続するためのインターフェースと、

記憶手段と、

前記インターフェースと前記記憶手段との間のデータ転送を制御するとともに、前記第1の計算機からの前記記憶手段へのデータ書き込みと、前記第2の計算機からの前記記憶手段のデータ読み出しとを並行して受け付け、前記記憶手段を、前記第1の計算機から第2の計算機への先入れ先出し(FIFO)機構として動作させる制御手段とを備えたことを特徴とする記憶装置サブシステム。

#### 【請求項 2】

請求項1に記載の記憶装置サブシステムにおいて、

前記制御手段は、前記第2の計算機からの前記記憶手段へのデータ書き込みと、前記第1の計算機からの前記記憶手段のデータ読み出しとを並行して受け付け、前記記憶手段を、前記第1の計算機と第2の計算機との間の先入れ先出し(FIFO)機構として動作せるものであるこ

[Claim(s)]

[Claim 1]

interface in order to connect with first computer and second computer respectively and,

storage means and,

As data transfer between aforementioned interface and aforementioned storage means is controlled, storage device subsystem . which designates that it has control means which in parallel, accepts data writing to aforementioned storage means from aforementioned first computer , and data reading of aforementioned storage means from aforementioned second computer operates aforementioned storage means , as first-in, first-out (FIFO ) mechanism to second computer from aforementioned first computer asfeature

[Claim 2]

In storage device subsystem which is stated in Claim 1 ,

storage device subsystem . which designates that it is something where the aforementioned control means , in parallel, accepts data writing to the aforementioned storage means from aforementioned second computer , and data reading of aforementioned storage means from aforementioned first computer operates the aforementioned

とを特徴とする記憶装置サブシステム。

#### 【請求項 3】

請求項 1 または 2 に記載の記憶装置サブシステムにおいて、

前記制御手段は、前記第 1 の計算機または第 2 の計算機からの前記記憶手段に対するアクセスを、シーケンシャルアクセスに限定することを特徴とする記憶装置サブシステム。

#### 【請求項 4】

請求項 1 または 2 に記載の記憶装置サブシステムにおいて、

前記制御装置は、前記第 1 および第 2 の計算機に対して、前記記憶手段の容量を実際の容量よりも大きな容量として見せ、前記第 1 および第 2 の計算機から前記記憶手段への実際の容量を越える領域へのアクセスを受け付けることを特徴とする記憶装置サブシステム。

#### 【請求項 5】

請求項 2 に記載の記憶装置サブシステムにおいて、

前記インターフェースは、カウントキーデータ形式のデータを受け付けるインターフェースと、固定長ブロック形式のデータを受け付けるインターフェースとを含み、

前記第 1 の計算機は、前記カウントキーデータ形式のデータを受け付けるインターフェースを介して、前記記憶手段へのアクセスをカウントキーデータ形式に従って行い、

前記第 2 の計算機は、前記固定長ブロック形式のデータを受け付けるインターフェースを介して、前記記憶手段へのアクセスを固定長ブロック形式に従って行い、

前記制御装置は、前記第 1 の計算機が前記記憶手段に対して書き込んだカウントキーデータ形式のデータを前記第 2 の計算機が読み出す際に、データ部分のみの固定長ブロック形式のデータとして読み出させ、または前記第 2 の計算機が前記記憶手段に対して書き込んだ固定長ブロック形式のデータを前記第 1 の計算機が読み出す際に、カウントキーデータ形式のデータとして読み出させることを特徴とする記憶装置サブシステム。

storage means , as aforementioned first computer and first-in, first-out (FIFO ) mechanism between second computer as feature

#### [Claim 3]

In storage device subsystem which is stated in Claim 1 or 2,

As for aforementioned control means , storage device subsystem . which designates that the access for aforementioned first computer or aforementioned storage means from second computer , is limited in sequential access as feature

#### [Claim 4]

In storage device subsystem which is stated in Claim 1 or 2,

storage device subsystem . which designates that access to region which shows theaforementioned controller , exceeds actual capacity to aforementioned storage means from computer of aforementioned first and second vis-a-vis computer of theaforementioned first and second , capacity of aforementioned storage means incomparison with actual capacity as large capacity , is accepted as feature

#### [Claim 5]

In storage device subsystem which is stated in Claim 2 ,

As for aforementioned interface , including interface which accepts the data of count key data form and interface which accepts data of fixed length block type,

As for aforementioned first computer , through interface which accepts the data of aforementioned count key data form , following access to theaforementioned storage means to count key data form , action,

As for aforementioned second computer , through interface which accepts the data of aforementioned fixed length block type, following access to theaforementioned storage means to fixed length block system, action,

As for aforementioned control device , aforementioned first computer occasionwhere aforementioned second computer reads out data of count key data form whichwas written vis-a-vis aforementioned storage means , as data of fixed length block type only of data portion reading out, or aforementioned second computer theoccasion where aforementioned first computer reads out data of the fixed length block type which was written vis-a-vis aforementioned storage means , It reads out as data of count key data form \* storage device subsystem . which designatesthing as feature

## 【請求項 6】

請求項 2 または 5 に記載の記憶装置サブシステムにおいて、

前記制御装置は、前記第 1 または第 2 の計算機が前記記憶手段に対して書き込むデータの文字コードを変換して書き込み、または前記第 1 または第 2 の計算機が前記記憶手段からデータを読み出す際に文字コードを変換して読み出すように制御することを特徴とする記憶装置サブシステム。

## Specification

## 【発明の詳細な説明】

## 【0001】

## 【発明の属する技術分野】

本発明は、情報処理システムなどに用いる記憶装置サブシステムに関し、特に、複数のデータ形式に従うインターフェースを有する記憶装置サブシステムに関する。

## 【0002】

## 【従来の技術】

近年、パソコンやワークステーションなどの小型計算機を使用して、従来メインフレームで行われてきた業務を行う「ダウンサイ징」が盛んに行われている。

メインフレームには、従来から行われてきた業務により大量の情報が蓄積されており、メインフレームで蓄積された情報を小型コンピュータからもアクセスしたいという要求がある。

従来、複数のコンピュータシステム間のデータ転送に関する问题是、IEEE( Institute of Electrical and Electronics Engineers ) 802 で規定されているイーサネット(Ethernet)が広く使われている。

## 【0003】

## 【発明が解決しようとする課題】

イーサネットは、さまざまな計算機で採用されているため、異機種間のデータ転送で広く使用されているが、ネットワークには多数の計算機が繋がるため、2 つの計算機間での大量データ転送は好ましくない。

さらにネットワークの転送速度は比較的低速であることも問題となる。

## 【0004】

## [Claim 6]

In storage device subsystem which is stated in Claim 2 or 5 ,

As for aforementioned controller , computer of aforementioned 1 st or 2nd converting character code of data which is written vis-a-vis the aforementioned storage means converting character code occasion where computer of writing , or aforementioned 1 st or 2nd reads out data from the aforementioned storage means , as read out, storage device subsystem . which designates that it controls as feature

## [Description of the Invention]

## [0001]

## [Technological Field of Invention]

this invention regards storage device subsystem which is used for information handling system etc, especially, it regards storage device subsystem which possesses interface which you follow data form of plural .

## [0002]

## [Prior Art]

Recently, using personal computer and workstation or other miniature computer , "downsizing " which does business which is done until recently with mainframe is done actively.

information of large scale compilation is done, to mainframe by business which is done from until recently, with mainframe information which the compilation is done disliking which access is done \* is request even from miniature computer .

Until recently, IEEE  
(Institute of Electrical and Electronics Engineers ) Ethernet  
(Ethernet ) which is stipulated with 802 is widely used in regard to data transfer between computer system of plural .

## [0003]

## [Problems to be Solved by the Invention]

Ethernet , because it is adopted with various computer , is widely used with data transfer between differing machine type , but because multiple computer is connected in network , large scale data transfer between 2 computer is not desirable.

Furthermore forwarding rate of network being a low speed relatively becomes problem .

## [0004]

また、複数のインターフェースを備えるディスク装置を介して、複数の計算機間でデータをやりとりすることによりデータ転送が実現できるが、通常のディスク装置にはホスト間のデータ転送に必要な機能は備わっていない。

例えば、ホスト A とホスト B に共有されるディスク装置 C がある場合、ホスト A がディスク装置 C にデータを書き込んでも、ホスト B はディスク装置 C のどの位置のデータを読めば良いか認識できないため、ディスク装置 C を介したデータ転送是不可能である。

#### 【0005】

本発明の目的は、従来システム間のデータ転送に使用していたネットワークに替えて、ディスク装置に代表される記憶装置を介したデータ転送を行うことのできる記憶装置サブシステムを提供することにある。

また本発明は、複数のインターフェースを備えるディスク装置に対し、ホスト A がデータを書き込み、ホスト B はその書き込まれたデータを、ホスト A の書き込み順通りに適切に読み出せる、先入れ先出し機構(FIFO)を備えた記憶装置サブシステムを提供することを目的とする。

#### 【0006】

##### 【課題を解決するための手段】

上記課題を解決するための記憶装置サブシステムの構成を以下に示す。

本発明における記憶装置サブシステムは、第 1 の計算機および第 2 の計算機に接続するためのインターフェースと、記憶手段(ディスクや半導体メモリなど)と、前記インターフェースと記憶手段との間で、データ転送を制御する制御装置を持つ。

制御装置では、前記第 1 の計算機からの前記記憶手段へのデータ書き込みと、前記第 2 の計算機からの前記記憶手段のデータ読み出しとを並行して受け付け、前記記憶手段を、前記第 1 の計算機から第 2 の計算機への先入れ先出し(FIFO)機構として動作させる。

さらに、制御手段は、前記第 2 の計算機からの前記記憶手段へのデータ書き込みと、前記第 1 の計算機からの前記記憶手段のデータ読み出しとを並行して受け付け、前記記憶手段を、前記第 1 の計算機と第 2 の計算機との間の先入れ先出し(FIFO)機構として動作させるようにしてもよい。

In addition, through disk drive which has interface of plural, it can actualize data transfer by exchanging data between computer of plural, but, as for function which is necessary for data transfer between host it has not been provided to conventional disk drive.

When there is a disk drive C which is shared to for example host A and host B, the host A writing data to disk drive C, as for host B you read the data of which location of disk drive C, whether it is good, because you cannot recognize, is through disk drive C data transfer is impossible.

#### 【0005】

Changing into network which has been used for data transfer between the system until recently, storage device which is represented in disk drive it is through objective of this invention, it is to offer storage device subsystem which can do data transfer.

In addition as for this invention, host A data that data which was written, according to writing order of host A reads out the writing, host B appropriately vis-a-vis disk drive which has interface of the plural, \*, it designates that storage device subsystem which has first-in, first-out mechanism (FIFO) is offered as objective.

#### 【0006】

##### 【Means to Solve the Problems】

configuration of storage device subsystem in order to solve above-mentioned problem is shown below.

storage device subsystem in this invention, interface and storage means in order to connect to first computer and second computer (Such as disc and semiconductor memory) with, aforementioned interface and between storage means, has control device which controls data transfer.

With controller, in parallel, it accepts data writing to aforementioned storage means from aforementioned first computer, and data reading of aforementioned storage means from aforementioned second computer it operates aforementioned storage means, as first-in, first-out (FIFO) mechanism to second computer from aforementioned first computer.

Furthermore, control means, in parallel, accepts as aforementioned first computer and first-in, first-out (FIFO) mechanism between second computer to operate is possible data writing to aforementioned storage means from aforementioned second computer, and data reading of aforementioned storage means from aforementioned first computer aforementioned storage means.

## 【0007】

そのような FIFO 機構として動作させるためには、前記インターフェースを介して接続される第 1 または第 2 の計算機からの記憶手段に対するアクセスは、シーケンシャル書き込み、あるいはシーケンシャル読み出しのみに限定する。すなわち、第 1 および第 2 の計算機から当該記憶手段に対して連続して書き込みないし読み出し指示が来た場合に、それぞれの指示に付随する書き込みアドレスおよび読み出しアドレスがシーケンシャルになるようにするとよい。

また、制御手段は、前記記憶手段が第 1 および第 2 の計算機に対して実際のディスク容量よりも大きいディスクに見えるようにする。

例えば、第 1、第 2 の計算機から容量問い合わせのコマンド(SCSI における modesense など)が来ると、記憶装置サブシステムは、自装置がもつ記憶手段例えばディスク装置の容量よりも大きな容量を返す。

さらに、実際に書き込み要求が来た場合には、シーケンシャルに書き込みていき、書き込み位置が実際の記憶手段の容量を越える位置になつたら、記憶手段の先頭に戻り、そこから再び書き込みを継続する。

同様に、記憶手段に対してシーケンシャル読み出しを行っていた場合、読み出し位置が実際に記憶手段の容量を越える位置になつたら、記憶手段の先頭に戻ってそこから再び読み出しを継続する。

このようにすることで、実際よりも大きな容量の記憶手段として動作する。

## 【0008】

また、第 1 の計算機が書き込みを行って、実際の記憶手段への書き込み位置が先頭に戻ったとき、以前書かれたデータがまだ第 2 の計算機から読み出されていない場合には、以前書かれたデータが第 2 の計算機から読み出されるまで第 1 の計算機からの書き込みはさせない。

逆に、第 2 の計算機が読み出しを行うとき、ディスク読み出し位置に第 1 の計算機からデータが書き込まれていない場合には、第 2 の計算機からの読み込みはさせない。

これにより、本記憶装置サブシステムは、第 1 の計算機と第 2 の計算機との間の先入れ先出し(FIFO)機構として動作する。

## 【0009】

## [0007]

In order to operate as that kind of FIFO mechanism , through aforementioned interface , limits access for storage means from computer of 1 st or 2nd whichis connected, in only sequential writing , or sequential reading , continuing from the computer of namely, first and second vis-a-vis this said storage means , when writing or reading indication comes, writing address and reading address which are annexed to indication of eachone that should have tried become sequential .

In addition, aforementioned storage means try control means , to be visible inlarge disc in comparison with actual disc capacity vis-a-vis computer of first and second .

When command (In SCSI such as mode sense ) of capacity inquiry comes from for example first , second computer , storage device subsystem returns large capacity in comparison with capacity of storage means for example disk drive which our device has.

Furthermore, when writing request comes actually, writing it does in sequential and when it becomes position where writing location exceeds the capacity of actual storage means , it returns to head of storage means , continues writing again from there.

When in same way, sequential reading was done vis-a-vis storage means , when itbecomes location where read position exceeds capacity of storage means actually, returning to head of storage means , it continues reading again from there.

By fact that it makes this way, it operates in comparison as storage means of large capacity with fact.

## [0008]

In addition, first computer doing writing , when writing location to actual storage means returns to head , in past data where data which iswritten still when it is not read out, in past is written from the second computer until it reads out, from second computer does not do writing from first computer .

When conversely, second computer does reading , when in disc read position data is not written from first computer , it does not read from second computer .

Because of this, this storage device subsystem operates as first-in, first-out (FIFO ) mechanism between the first computer and second computer .

## [0009]

また、記憶手段を、第 1 の計算機からは書き込みのみが可能で前記第 2 の計算機からは読み出しのみが可能なボリュームに見せかける、または第 1 の計算機からは読み出しのみが可能で第 2 の計算機からは書き込みのみが可能なボリュームに見せかけることで、FIFO 機構として動作しているボリュームに不当なアクセスが行われないようにしても良い。

#### 【0010】

さらに、この記憶装置サブシステムのインターフェースが、カウントキーデータ形式のデータを受け付けるインターフェースと、固定長ブロック形式のデータを受け付けるインターフェースとが存在する場合でも FIFO 機能を実現する。

第 1 の計算機は、記憶装置サブシステムへのアクセスをカウントキーデータ形式で行い、第 2 の計算機は記憶装置サブシステムへのアクセスを固定長ブロック形式に従って行うとする。

そのとき、制御装置は、第 1 の計算機が前記記憶手段に対して書き込んだカウントキーデータ形式のレコードを第 2 の計算機が読み出す際に固定長ブロック形式のデータとして、データ部分のみを読み出させる機能を持つ。

または第 2 の計算機が記憶手段に対して書き込んだ固定長ブロック形式のデータを第 1 の計算機が読み出す際にはカウントキーデータ形式のデータに変換して読み出せる機能を持たせる。

これにより、データ形式が異なる計算機間での FIFO 機能を実現することができる。

また、カウントキーデータ形式と固定長ブロック形式のデータ変換と同時に、文字コード変換、例えば ASCII コードと EBCDIC コードの相互変換を行わせるなどの機能を附加させることも可能である。

#### 【0011】

##### 【発明の実施の形態】

以下、図面を用いて本発明の実施の形態を説明する。

#### 【0012】

図 1 に、本発明の第 1 の実施の形態に係る計算機システムの構成例を示す。

この計算機システムは、ホスト 1、ホスト 2、そしてホスト 1 とホスト 2 に接続される記憶装置サブシステム 3 から構成される。

In addition, storage means , only writing being possible from first computer ,from aforementioned second computer only reading camouflages in thepossible volume , or only reading being possible from first computer , from second computer by fact that only writing camouflages in possible volume , That unjust access is not done in volume which operates as the FIFO mechanism it is good.

#### [0010]

Furthermore, interface of this storage device subsystem , actualizes FIFO function even withwhen interface which accepts data of count key data form and interface which accepts data of fixed length block type exist.

first computer does access to storage device subsystem with count key data form , as for second computer following access to storage device subsystem to fixed length block system, assume that it does.

That time, controller first computer reads out only data portion occasionwhere second computer reads out record of count key data form which was writtenvis-a-vis aforementioned storage means as data of fixed length block type, \* it has function.

Or second computer case where first computer reads out data of fixed length block type which was written vis-a-vis storage means converting to data of count key data form , it reads out, \* it can give function.

Because of this, data form between different computer , FIFO function can be actualized.

In addition, simultaneously with data conversion of count key data form and fixed length block system, also it is possible to add or other performance which converts character code conversion, for example ASCII code and EBCDIC code mutually.

#### [0011]

##### [Embodiment of the Invention]

embodiment of this invention is explained below, making use of drawing .

#### [0012]

In Figure 1 , configuration example of computer system which relates to first embodiment of the this invention is shown.

this computer system configuration is done from storage device subsystem 3 which is connected to the host 1, host 2, and host 1 and host 2.

## 【0013】

記憶装置サブシステム 3 は、ホスト 1 と接続するインターフェース 31、ホスト 2 と接続するインターフェース 32、ディスク 33、制御用メモリ 34、制御装置 35、および設定用コンソール 36 から構成される。

ディスク 33 は複数存在することもできる。

## 【0014】

なお、ホストコンピュータ 1,2 と記憶装置サブシステム 3 とのインターフェースはここでは固定長インターフェース(固定長形式)とする。

固定長形式では、各データはブロックと呼ばれる領域に格納される。

各ブロックは例えば 512 バイトなどの固定長である。

各ブロックには、ブロック番号(以下、LBA と言う)と転送長を指定してアクセスする。

転送長は固定長ブロックの数を指定する。

## 【0015】

記憶装置サブシステム 3 は、設定用コンソール 36 から入力された指示に応じてディスク 33 の役割を変える機能を持つ。

ディスク 33 の役割としては、ホスト 1 専用のデバイスになる場合、ホスト 2 専用のデバイスになる場合、そしてホスト 1 およびホスト 2 の両方から使用できる共有デバイスになる場合がある。

共有デバイスになる場合には、さらにホスト 1 とホスト 2 との間でのデータ転送に使用できる FIFO デバイスにもなる。

## 【0016】

ディスク 33 が FIFO デバイスに設定された場合について説明する。

FIFO デバイスでは、ホストから見た場合、実際のディスク 33 の容量を上回る極めて大きいディスクとして見える。

例えば、インターフェース 31 およびインターフェース 32 が SCSI インターフェースの場合、ホスト 1 およびホスト 2 から記憶装置サブシステム 3 に対して modesense コマンドを出すと、容量が返却される。

ディスク 33 が FIFO デバイスに設定された場合は、ディスク 33 の実際の容量よりも大きな容量を示す値が返却される。

## [0013]

storage device subsystem 3 configuration is done from memory 34, control device 35, for interface 32, disc 33, control and the console 36 for setting which are connected with interface 31 , host 2 which is connected with host 1.

disc 33 can also exist plural .

## [0014]

Furthermore, interface of host computer 1, 2 and storage device subsystem 3 does here fixed length interface (fixed length form ) with.

With fixed length form , as for each data it is housed in region which is called block .

Each block is for example 512byte or other fixed length .

block number (Below, you call LBA ) with appointing transfer length, access it does in each block .

Transfer length appoints quantity of fixed length block .

## [0015]

storage device subsystem 3 has function which changes role of disc 33 according to indication which is inputted from console 36 for setting.

As role of disc 33, when it becomes device of host 1 dedicated , when it becomes device of host 2 dedicated , and there are times when it becomes joint ownership device which can be used from both of host 1 and host 2.

When it becomes joint ownership device , furthermore it becomes also FIFO device which can be used for data transfer between host 1 and the host 2.

## [0016]

When disc 33 is set to FIFO device , being attached, you explain.

It is visible quite as large disc where with FIFO device , when you saw from host , capacity of actual disc 33 is exceeded.

When for example interface 31 and interface 32 are SCSI interface , when mode sense command is put out from host 1 and host 2 vis-a-vis storage device subsystem 3, capacity is returned.

When disc 33 is set to FIFO device , value which shows large capacity in comparison with actual capacity of disc 33 is returned.

また、ホストからはシーケンシャルアクセスのみを可能とし、不連続なアクセスはできない。

アクセスは必ず先頭から実施されなければならず、それ以降は順番にアクセスされる必要がある。

このための実装としては、ホストから来たコマンドが不連続なアクセスの場合に、処理失敗としてホストに返す方法、または記憶装置サブシステム3側でホストが指定したアドレスを一切無視し、到着したデータをシーケンシャルにアクセスする方法とがあり得る。

以下、本実施の形態では後者の実装の場合について説明していく。

また、FIFOデバイス33に対する不当なアクセスを防ぐため、ホスト1からは書き込み専用でホスト2からは読み出し専用にする、あるいはホスト1からは読み出し専用でホスト2からは書き込み専用にする機能もある。

#### 【0017】

図2で、制御用メモリ34について説明する。

制御用メモリ34には、FIFOデバイス用のテーブル40を持ち、該テーブル40には各LBAに対応するフラグ41を保有する。

なお、フラグ41を保有するLBAは、ホストからディスク33を実際の容量よりも大きな容量のディスクとして見た場合のLBAではなく、実際のディスク33上のLBAである。

あるLBAのアドレス位置に書き込みが来ると、テーブル40の該LBAに対応するフラグ41を1に設定する。

そして、別のホストからそのLBAのアドレスの内容が読み出されると、テーブル40の該フラグ41を0に戻す。

なお、本実施の形態ではLBA単位にフラグを持つが、それ以外にも複数個のブロック単位にフラグを持って管理する方式でも良い。

#### 【0018】

さらに、制御用メモリ34には、リードアドレステーブル42とライトアドレステーブル43を保有する。

それぞれ初期値は0である。

リードアドレステーブル42はFIFOデバイス33から読み出しそるときの読み出し開始LBAを示し、ライトアドレステーブル43はFIFOデバイス33からの書き込み開始LBAを示す。

In addition, it makes only sequential access possible from host, the discontinuous access is not possible.

access does not become, if it is not executed by all means from the head, after that is necessary access to be done in sequence.

As mount because of this, when command which comes from the host is discontinuous access, method of returning to host as the failure of treatment. Or address which host appoints on storage device subsystem 3 side is altogether ignored, data which arrives method which access is done is impossible in sequential.

Below, with this embodiment in case of mount of the latter being attached, you explain.

In addition, in order to prevent unjust access for FIFO device 33, from host 1 with writing dedicated from host 2 it makes read-only, or from host 1 with read-only there is also a function which is made the writing dedicated from host 2.

#### [0017]

With Figure 2, you explain concerning memory 34 for control.

In memory 34 for control, flag 41 which corresponds to each LBA is possessed in said table 40 with table 40 of FIFO device use.

Furthermore, LBA which possesses flag 41 is not LBA when you saw disc 33 in comparison with actual capacity as disc of the large capacity from host is LBA on actual disc 33.

When writing comes to address location of a certain LBA, flag 41 which corresponds to said LBA of table 40 is set to 1.

When and, content of address of LBA reads out from another host, said flag 41 of table 40 is reset to 0.

Furthermore, with this embodiment it has flag in LBA unit, but other than that and it is good to block unit of plurality with system which is managed with flag.

#### [0018]

Furthermore, read address table 42 and write address table 43 are possessed in memory 34 for control.

initial value is 0 respectively.

read address table 42 when reading doing from FIFO device 33, shows reading start LBA, write address table 43 when writing doing in FIFO device 33, shows writing start LBA.

33 に書き込みするときの書き込み開始 LBA を示す。

これらの LBA も、ホストからディスク 33 を実際の容量よりも大きな容量のディスクとして見た場合の LBA ではなく、実際のディスク 33 上の LBA である。

#### 【0019】

図 3 のフローチャートを参照して、FIFO デバイスに対する書き込みが来た場合の記憶装置サブシステム 3 の動作について説明する。

まず、FIFO デバイスに対する書き込みおよび読み出しを開始する前に、FIFO デバイスの初期化を行う。

初期化では、制御装置 35 が制御メモリ 34 の内容、ライトアドレステーブル 43、およびリードアドレステーブル 42 の内容を全て 0 にして、ディスク装置 33 に何も書かれていない状態を作る。

初期化は、設定用コンソール 36 から FIFO デバイスに対して初期化の命令を出すか、ホストから初期化用のコマンドを FIFO デバイスに出すことで行う。

#### 【0020】

ホストからの書き込み要求が来ると、制御装置 35 はライトアドレステーブル 43 を見て、書き込みを実施する LBA を決定する(ステップ 101)。

次に、その LBA に対応するフラグ 41 の内容を参照し、0 かどうかチェックする(ステップ 102、103)。

0 の場合にはその LBA の位置に書き込みをしても良いということだから、次のステップ 104 に進む。

該フラグが 1 の場合には、0 になるまで待ち続ける。

ステップ 104 ではライトデータを対象 LBA に書き込み、ステップ 105 では今回書き込んだ当該 LBA に対応するテーブル 40 内のフラグ 41 を 1 にする。

ステップ 106 ではライトアドレステーブル 43 に 1 加算する。

もし、ライトアドレステーブル 43 に記載されている LBA がディスク 33 の最大ブロック数以上になつた場合には、ライトアドレステーブル 43 の内容を 0 に戻す(ステップ 107、108)。

これにより、ディスク 33 内の最終ブロックまで書き込んだ場合、次の書き込み LBA は 0、すなわ

Either these LBA, are not LBA when you saw disc 33, in comparison with actual capacity as disc of large capacity from the host are LBA on actual disc 33.

#### [0019]

Referring to flowchart of Figure 3 , you explain concerning the operation of storage device subsystem 3 when writing for FIFO device comes.

First, before starting writing and reading for FIFO device , the initialization of FIFO device is done.

With initialization , control device 35 state which is not written on disk drive 33 at all with content , write address table 43, of control memory 34 and content of read address table 42 as all 0, is made.

initialization , it puts out instruction of initialization from console 36 for setting vis-a-vis FIFO device , or it does by fact that command for initialization is put out to FIFO device from host .

#### [0020]

When writing request from host comes, control device 35 looking at the write address table 43, decides LBA which executes writing (step 101) .

Next, you refer to content of flag 41 which corresponds to the LBA, whether or not 0 check you do, (step 102, 103) .

In case of 0 therefore notion that where, it is good to location of LBA doing writing , it advances to following step 104.

When said flag 1 is, until it becomes 0, it continues to wait.

With step 104 write data in object LBA with writing , step 105 flag 41 inside table 40 which corresponds to this said LBA which this time was written is designated as 1.

With step 106 1 it adds to write address table 43.

When LBA which is stated in write address table 43 is above quantity of maximum block of disc 33, content of write address table 43 is reset to 0, (step 107, 108) .

Because of this, when you wrote to final block inside disc 33, the following writing LBA 0, namely becomes head of disc

ちディスク 33 の先頭となる。

【0021】

次に、ホストからの読み出し要求が来たときの記憶装置サブシステム 3 の動作について、図 4 のフローチャートを参照して説明する。

【0022】

最初にブロックの読み出し要求が来ると、制御装置 35 はリードアドレステーブル 42 を見て読み出し対象の LBA を決定する(ステップ 201)。

次にテーブル 40 内の読み出し対象 LBA のフラグ 41 を参照し、1 かどうかチェックする(ステップ 202、203)。

1 の場合にはその LBA のブロックに読み出すべきデータがあるということだから、次のステップ 204 に進む。

0 の場合には 1 になるまで待ち続ける。

ステップ 204 では対象 LBA のブロックを読み出し、ステップ 205 では今回読み出した LBA に対するテーブル 40 内のフラグ 41 を 0 にする。

ステップ 206 でリードアドレステーブル 42 に 1 加算する。

もし、リードアドレステーブル 42 に記載されている LBA がディスク 33 の最大ブロック数以上になつた場合にはリードアドレステーブル 42 の内容を 0 に戻す(ステップ 207、ステップ 208)。

これにより、ディスク 33 内の最終ブロックまで読み出した場合、次の読み出し LBA は 0、すなわちディスク 33 の先頭となる。

【0023】

以上の動作を、例えばホスト 1 では書き込み、ホスト 2 では読み出しを行うように実行すると、ホスト 2 はホスト 1 がディスク 33 に書き込んだ内容を先頭から順に読み出すことができる。

また、例えばホスト 1 からディスク 33 上の全領域にデータを書き込むと、次の書き込み位置は先頭に移されるが、書き込み前にテーブル 40 を参照しフラグ 41 をチェックすることで、ホスト 2 がまだ読み出していないアドレスについての書き込みを抑止できる。

ホスト 2 の読み出しが完了するとホスト 1 は再び先頭から書き込みを行い、ホスト 2 はディスク 33 の全領域を読み終わるとディスク 33 の先頭から読み始める。

これにより、ディスク 33 は FIFO(先入れ先出し機

33.

[0021]

When next, reading request from host comes, referring to flowchart of Figure 4 concerning operation of storage device subsystem 3, you explain.

[0022]

When reading demand for block comes first, control device 35 looking at the read address table 42, decides LBA of reading object (step 201 ).

You refer to flag 41 of reading object LBA inside table 40 next, whether or not 1 check you do, (step 202, 203 ).

In case of 1 therefore notion that where, it is data which it should read out in block of LBA it advances to following step 204.

In case of 0 until it becomes 1, it continues to wait.

With step 204 block of object LBA with reading , step 205 this time the reading is flag 41 inside table 40 which corresponds to LBA is designated as 0.

It adds to read address table 42 with step 206.

When LBA which is stated in read address table 42 is above quantity of maximum block of disc 33 content of read address table 42 is reset to 0, (step 207, step 208 ).

Because of this, reading it is to final block inside disc 33 when, following reading LBA 0, namely becomes head of disc 33.

[0023]

Operation above, in order with for example host 1 with writing , host 2 to do the reading , when it executes, as for host 2 to read out from head in order it is possible content which host 1 wrote to disc 33.

In addition, when data is written to entire region on disc 33 from for example host 1, following writing location is moved to head , but table 40 is referred to before writing and by fact that check it does flag 41, writing host 2 still reading \* concerning address which is not can be controlled.

When reading of host 2 completes, host 1 does writing again from head , when entire region of disc 33 is read through, starts reading host 2 from head of disc 33.

Because of this, disc 33 operates FIFO (first-in, first-out

構:First In First Out)として動作する。

逆にホスト2からの読み込みがホスト1からの書き込みよりも早い場合も、ホスト1がまだ書き込んでいない領域へのホスト2からの読み出し要求は抑止でき、FIFOとして動作する。

#### 【0024】

上記の実施の形態では、ホストからの書き込みおよび読み出しにディスクを使用したが、一方のホストから書き込まれたデータは、他方のホストから読み出されるまで保存されれば良く、大容量の記憶装置は必要としない。

そのため、ディスクの代わりに、キヤッッシュメモリなどで使用される半導体メモリを使用しても実現可能である。

#### 【0025】

次に、本発明の第2の実施の形態を説明する。

#### 【0026】

図5に、本発明の第2の実施の形態に係る計算機システムの構成例を示す。

この計算機システムは、ホスト1、ホスト2、そしてホスト1とホスト2に接続される記憶装置サブシステム3から構成される。

#### 【0027】

記憶装置サブシステム3は、ホスト1と接続するCKDインターフェース1031、ホスト2と接続する固定長インターフェース1032、ディスク33、制御用メモリ34、制御装置35、および設定用コンソール36から構成される。

ディスク33は複数存在することもできる。

#### 【0028】

上述した第1の実施の形態で用いた固定長形式では、各データはブロックと呼ばれる領域に格納される。

各ブロックは、例えば512バイトなどの固定長である。

各ブロックにはブロック番号LBAと転送長を指定してアクセスする。

転送長は固定長ブロックの数を指定する。

#### 【0029】

一方、CKD形式では、シリンド番号(CC)、ヘッド番号(HH)、およびレコード番号(R)を指定してレコードにアクセスする。

mechanism :First In First Out ) as.

Case it is quick in comparison with writing from host 1 conversely reading from host 2, be able to control reading request from host 2 to region where host 1 still does not write, it operates as FIFO .

#### [0024]

With above-mentioned embodiment , disc was used for writing and reading from host , but data which was written from one hand host , until it reads out, from host of other if it is retained, to be good, storage device of large capacity does not need.

Because of that, using semiconductor memory which is used in place of disc ,with cache memory etc, it is a realizable .

#### [0025]

Next, second embodiment of this invention is explained.

#### [0026]

In Figure 5 , configuration example of computer system which relates to second embodiment of this invention is shown.

this computer system configuration is done from storage device subsystem 3 which is connected to the host 1, host 2, and host 1 and host 2.

#### [0027]

storage device subsystem 3 configuration is done from memory 34, control device 35, for fixed length interface 1032, disc 33, control and the console 36 for setting which are connected with CKDinterface 1031, host 2 which is connected with host 1.

disc 33 can also exist plural .

#### [0028]

With fixed length form which is used with first embodiment which description above is done, as for each data it is housed in region which is called block .

Each block is for example 512byte or other fixed length .

Appointing block number LBA and transfer length to each block , access it does.

Transfer length appoints quantity of fixed length block .

#### [0029]

On one hand, with CKDform , cylinder number (CC), head number (HH), and appointing record number (R), access it does in record .

アクセスの最小単位はレコードである。

以下、シリンド番号、ヘッド番号、レコード番号で表されるレコードアドレスを CCHHR と呼び、シリンド番号、ヘッド番号で表されるトラックのアドレスを CCHH と呼ぶ。

CKD 形式では、カウント部(以下、C 部と呼ぶ)、キー部(以下、K 部と呼ぶ)、データ部(以下、D 部と呼ぶ)で 1 つのレコードを形成する。

C 部には CCHHR と、K 部および D 部の長さが入り、常に固定長である。

K 部および D 部は可変長で、C 部に記された長さを有する。

レコードが複数集まって、1 つのトラックを形成するが、トラックの長さは固定長のため、各レコードの長さが異なる場合、1 トラックに入るレコード数はそれぞれのトラックによって異なる。

#### 【0030】

記憶装置サブシステム 3 は、設定用コンソール 36 から入力された指示に応じてディスク 33 の役割を変える機能を持つ。

ディスク 33 の役割としては、固定長データを格納する固定長ディスクになる場合、CKD データを格納する CKD ディスクになる場合、そしてデータ転送用の FIFO デバイスになる場合がある。

FIFO デバイスになる場合は、CKD インタフェース 1031 と固定長インターフェース 1032 からの両方からアクセスできる。

ディスク 33 が複数存在する場合には、そのそれについて設定用コンソール 36 から機能を設定できる。

#### 【0031】

図 6A は一般的な CKD ディスク装置におけるトラック上のデータ配置を示し、図 6B は CKD 形式のデータを本実施形態の記憶装置サブシステム 3 に格納する場合のデータ形式を示す。

本実施の形態において、ディスク 33 は物理的には固定長データ形式に従うものを使用し、CKD ディスクとして動作する場合、CKD 形式のデータを固定長データ形式に変換して格納している。

#### 【0032】

図 6A の HA(51) はホームアドレスで、トラックの状態などを示す。

R0C(52) はレコード 0 のカウント部、R0D(53) はレコード 0 のデータ部で、ここにはユーザデータは

minimum unit of access is record .

Below, record address which is displayed with cylinder number , head number , record number is called the CCHHR, address of track which is displayed with cylinder number , head number is called CCHH.

With CKDform , count section (Below, it calls C part ), key part (Below, it calls K section), record of the one is formed with data part (Below, it calls D part ).

CCHHR and K section and length of D part enter in C part , it is a normally fixed length .

K section and D part with variable length , have length which was inscribed to C part .

record getting together plural , track of one is formed, but as for length of track because of fixed length , the length of each record in case of different , as for quantity of record which enters into 1 track with respective track different .

#### 【0030】

storage device subsystem 3 has function which changes role of disc 33 according to indication which is inputted from console 36 for setting.

As role of disc 33, when it becomes fixed length disc which houses the fixed length data , when it becomes CKDdisc which houses CKDdata , and there are times when it becomes FIFO device for data transfer .

When it becomes FIFO device , access it is possible from both from CKDinterface 1031 and fixed length interface 1032.

When disc 33 plural it exists, function can be set from console 36 for setting that concerning respectively.

#### 【0031】

As for Figure 6A data arrangement on track in general CKDdisk drive is shown, Figure 6B shows data form when data of CKDform is housed in the storage device subsystem 3 of this embodiment .

In this embodiment , disc 33 uses those which you follow fixed length data form to the physical , when it operates as CKDdisc , converting data of the CKDform to fixed length data form , has housed.

#### 【0032】

HA (51) of Figure 6A with home address , shows state etc of the track .

As for R0C (52) count section of record 0, as for R0D (53) with data part of record 0, as for user data it is not

通常格納されない。

R1C(54)、R1D(55)はそれぞれレコード 1 のカウント部、データ部である。

それぞれのフィールドの間にはギャップと呼ばれるデータの格納されない領域が存在し、それぞれの区切りとなっている。

C 部の内容、ギャップはディスク装置内部で作られる。

#### 【0033】

記憶装置サブシステム 3 では、全トラックの HA(51)、R0C(52)、R0D(53)を別領域にまとめて格納し、固定長インターフェース 33 を介しては、HA(51)、R0C(52)、R0D(53)は見えないようにしてある。

記憶装置サブシステム 3 に CKD インタフェース 1031 から書き込みが来た場合には、制御装置 35 により、図 6B のように各レコードの C 部、D 部を固定長ブロックの先頭から前詰めにしてディスク 33 に格納していく。

また、レコードとレコードの間には空白にし、図 6B の例では LBA2(62)のレコード 1 の終端に空き領域があるが、レコード 2 は LBA3(63)から始まる。

C 部と D 部の間に K 部(キー)が存在することもあるが、ここではキーなしレコードについて考慮し、K 部は存在しないこととする。

#### 【0034】

次に、ディスク 33 が FIFO デバイスに設定された場合について説明する。

FIFO デバイスでは、決められたサイズのデータしかアクセスできない。

例えば CKD インタフェース 1031 からのアクセスは、通常ならば任意の大きさのレコードを読み書きできるが、FIFO デバイスとしての使用の場合は、サイズは 1 種類で、かつ固定長ブロックの大きさの倍数とする。

また、固定長インターフェース 1032 からのアクセスも、一種類のサイズに限定し、CKD インタフェース 1031 からアクセスされるレコードサイズと同じとする。

さらに、FIFO デバイスは、ホストからはシーケンシャルアクセスのみを可能とし、不連続なアクセスは禁止される。

アクセスは必ず先頭から実施され、それ以降は順番にアクセスされる必要がある。

housedusually here.

R1C (54), R1D (55) count section of respective record 1,is data part .

region where data which is called gap is not housedexists between respective field , has become respective partition .

content , gap of C part is made with disk drive interior .

#### [0033]

With storage device subsystem 3, HA of all track (51), R0C (52), collecting R0D (53) to another region , it houses, through fixed length interface 33, HA (51), R0C (52), as for R0D (53) try not to be visible.

When writing comes from CKDinterface 1031 in storage device subsystem 3, with control device 35, like Figure 6B it houses in disk 33 from head of fixed length block C part , D part of each record to front stuffing.

In addition, it designates between record and record as the blank , with example of Figure 6B there is an empty region in terminal of record 1 of LBA2 (62), but record 2 starts from LBA3 (63).

Also fact that K section (key ) exists between C part and D part it is, but here it considers concerning key none record , K section we do not exist.

#### [0034]

When next, disc 33 is set to FIFO device , being attached, you explain.

With FIFO device , only data of size which is decided access it is possible.

But as for access from for example CKDinterface 1031, if it is usual, read-write ispossible record of size of option , in case of use as the FIFO device , as for size with 1 kind , at same time it makes the number of multiples of size of fixed length block .

In addition, it limits also access from fixed length interface 1032, in size of one kind , it makes same as record size which access is done from CKDinterface 1031.

Furthermore, FIFO device makes only sequential access possible from host ,discontinuous access is prohibited.

access is executed by all means from head , after that isnecessary access to be done in sequence .

また、ホストから見た場合、実際のディスク33の容量を上回る極めて大きいディスクとして見える。

#### 【0035】

FIFO デバイスに対する書き込みが来た場合について説明する。

CKD インタフェース 1031 から書き込みが来た場合には、ホスト 1 から C 部および K 部(K 部が存在する場合)も渡され、制御装置 35 により C 部および D 部が図 6B のような形式に変換して格納される。

各レコードは先頭から順に書き込みが行われ、最終シリンダの最終トラックまでレコードが書き込まれた場合、その次に書き込まれたレコードは再び先頭に書き込まれる。

このとき、C 部には CKD インタフェース 1031 から送られてきたシリンダ番号およびヘッド番号が書き込まれる。

すなわち、実際にはディスク 33 の先頭にレコードが書き込まれるが、ホスト 1 側ではディスク 33 を実際よりも大きい容量を持つものとして認識しているため、位置情報は最終トラックの次のトラックを示すアドレスが来て、そのアドレスが記される。

以後、書き込まれるレコードの位置情報も同様に実際とは異なるアドレスが書き込まれる。

#### 【0036】

一方、固定長インターフェース 1032 からブロックの書き込みが来た場合には、LBA を指定して来るが、固定長インターフェース 1032 からは図 6B のように見えず、C 部や各レコードのギャップが取り除かれた図 6C のイメージで見える。

そのため、ホスト 2 からは、データの書かれているブロック番号は実際の LBA とは違うものが見えている。

以下図 6B の LBA と区別するため、ホスト 2 から見えるブロック番号はデバイスアドレス 71 と呼ぶ。

すなわち、ホスト 2 側からは、ディスク 33 を実際の容量を上回る極めて大きい容量のディスクとして見たときのデバイスアドレス 71 が与えられる。

#### 【0037】

制御装置 35 は、固定長インターフェース 1032 からデバイスアドレス 71 とデータを受ける。

In addition, when you saw from host , it is visible quite as the large disc where capacity of actual disc 33 is exceeded.

#### [0035]

When writing comes for FIFO device comes being attached, you explain.

When writing comes from CKDinterface 1031, also C part and K section (K section exists when ) are transferred from host 1, converting to form the C part and D part like Figure 6B , with control device 35 are housed.

Each record from head writing is done in order, when record was written to final track of final cylinder , record which that next waswritten is written to head again.

At time of this , cylinder number and head number which are sent from CKDinterface 1031 are written in C part .

record is written to head of disc 33 namely, actually, but because on, host 1 side you have recognized in comparison disc 33 as those which have large capacity with fact as for positional information address which shows following track of final track coming, address is inscribed.

From now on, different address is really written also positional information of record which is written in same way.

#### [0036]

On one hand, when writing of block comes from fixed length interface 1032, appoints LBA, from fixed length interface 1032 like Figure 6B , it is not visible,C part and it is visible with image of Figure 6 C where the gap of each record is removed.

Because of that, as for block number where data is written actual LBA those which are different have been visible from host 2.

In order to distinguish with LBA of below Figure 6B , block number which is visible from host 2 calls device address 71.

When seeing quite as disc of large capacity where, actual capacity is exceeded from namely, host 2 side device address 71 can give disc 33.

#### [0037]

control device 35 receives device address 71 and data from fixed length interface 1032.

それから、デバイスアドレス 71 から CCHHR を算出して、C 部を作成する。

FIFO デバイスでは、各レコードが固定長という制限を設けているので、デバイスアドレス 71 から CCHHR は容易に算出できる。

例えば、1 シリンダ内のトラック数を a、1 トラック内に収まるレコード数を b、ホストから与えられるデバイスアドレスを L とすると、

$$CC = L \div (a \times b)$$

$$HH = (L - (CC \times a \times b)) \div b$$

$$R = L - (CC \times a \times b) - (HH \times b)$$

で算出できる。

ただし、除算で小数部分は切り捨てる。

C 部作成後は、作成された C 部とともに D 部を図 6B のような CKD 形式のレコードとしてディスク 33 に格納する。

なお、上記式は、ディスク 33 を実際の容量を上回る極めて大きい容量のディスクとして見たときの変換だけでなく、実際のディスク 33 上のアドレスの変換にも用いることができる。

また、上記式の関係に基づき CCHHR からデバイスアドレスへの変換式も導き出せる。

#### 【0038】

さらに、最終ブロックまで書き込まれると、次に来たデータは CKD インタフェース 1031 からの書き込みと同様にディスク 33 の先頭へ書かれることになる。

このとき作成される C 部は、最終ブロックに付けられている CCHH の次のトラックを示すアドレスが付けられる。

#### 【0039】

図 7 で、制御用メモリ 34 の内容について説明する。

制御用メモリ 34 には、FIFO デバイス用のテーブル 40 を持ち、該テーブル 40 には各 CCHHR とそれに対応するフラグ 41 を持つ。

なお、フラグ 41 を保有する CCHHR は、ホストからディスク 33 を実際の容量よりも大きな容量のディスクとして見た場合の CCHHR ではなく、実際にディスク 33 上に割り当てられた図 6B のイメージでの CCHHR である。

Then, calculating CCHHR from device address 71, it draws up C part .

Because with FIFO device , restriction, each record fixed length is provided,it can calculate CCHHR easily from device address 71.

Quantity of track inside for example 1cylinder when device address which can give quantity of record which is settled inside a, 1track from b, host is designated as L,

$$CC=L \div (a \times b)$$

$$L - (CC \times a \times b) \div b HH=$$

$$R=L - (CC \times a \times b) - (HH \times b)$$

So it can calculate.

However, it cuts down small several parts amount with division.

After C part compilation, it houses in disc 33 with C part which was drawn up D part as record of CKD form like Figure 6B .

Furthermore, when seeing quite as disc of large capacity where above Formula disc 33 exceeds actual capacity not only a conversion, you can use for also conversion of address on actual disc 33.

In addition, you deduce also conversion equation to device address from CCHHR, on basis of relationship of above Formula \*.

#### [0038]

Furthermore, when it is written to final block , data which comes next in same way as writing from CKD interface 1031 means to head of disc 33 to be written.

As for C part which at time of this is drawn up, you can attach address which shows following track of CCHH which is attached to final block .

#### [0039]

With Figure 7 , you explain concerning content of memory 34 for control.

In memory 34 for control, each CCHHR it has flag 41 which corresponds to that in said table 40 with table 40 of FIFO device use .

Furthermore, CCHHR which possesses flag 41 is not CCHHR when you saw disc 33 in comparison with actual capacity as disc of the large capacity from host is CCHHR with image of Figure 6B which is allotted actually on disc 33.

ある CCHHR のレコードに対し書き込みが来ると、制御装置 35 はレコードが書き込まれた時点でテーブル 40 の当該 CCHHR に対応するフラグ 41 を 1 に設定する。

そして、別のホストからそのレコードの内容が全て読み出されると、当該フラグ 41 を 0 に戻す。

ここではレコード単位として説明しているが、トラックあるいはシリンダという大きな単位で管理しても良い。

#### 【0040】

さらに、制御用メモリ 34 には、リードアドレステーブル 42 とライトアドレステーブル 43 を保有する。

リードアドレステーブル 42、ライトアドレステーブル 43 は、それぞれ、FIFO デバイスから読み出しましたは書き込みするときのデバイスアドレス 71 または CCHHR を示す。

CKD インタフェース 1031 から書き込まれ、固定長インターフェース 1032 から読み出される場合には、ライトアドレステーブル 43 は CCHHR で管理され、初期値は(CC,HH,R)=(0,0,1)となる。

また、リードアドレステーブル 42 は、デバイスアドレス 71 の単位で管理され、初期値は 0 となる。

なお、これらのデバイスアドレス 71 と CCHHR も、ホストからディスク 33 を実際の容量よりも大きな容量のディスクとして見た場合のデバイスアドレスや CCHHR ではなく、実際にディスク 33 上に割り当てられたイメージでのデバイスアドレスや CCHHR である。

#### 【0041】

次に、FIFO デバイスに対する書き込み、読み出しが来たときの記憶装置サブシステム 3 の動作について説明する。

#### 【0042】

まず、図 8 のフローチャートを参照して、ホスト 1 から FIFO デバイスに対して書き込みを行ったときの処理を説明する。

FIFO デバイスに対する書き込み、読み出しを開始する前に、FIFO デバイスの初期化を行う。

初期化では、図 7 のテーブル 40 のフラグ 41 を全て 0 にして、ディスク装置 33 に何も書かれていない状態を作る。

また、リードアドレステーブル 42 はデバイスアドレスを記録するものとし、初期値は 0 とする。

When writing comes vis-a-vis record of a certain CCHHR, control device 35 sets flag 41 which corresponds to this said CCHHR of table 40 with the time point where record was written to 1.

When and, content of record all reads out from another host , this said flag 41 is reset to 0.

Here it is explanatory as record unit , it is good managing with the large unit , but track or cylinder .

#### [0040]

Furthermore, read address table 42 and write address table 43 are possessed in memory 34 for control.

read address table 42, write address table 43, when respectively, reading or writing doing from the FIFO device , shows device address 71 or CCHHR.

It is written from CKDinterface 1031, when it reads out from fixed length interface 1032, the write address table 43 is managed with CCHHR, initial value becomes (CC, HH, R ) = (0, 0 and 1) with.

In addition, read address table 42 is managed with unit of device address 71, the initial value becomes with 0.

Furthermore, either these device address 71 and CCHHR, are not device address and CCHHR when you saw disc 33 in comparison with actual capacity as the disc of large capacity from host are device address and CCHHR with image which is allotted actually on disc 33.

#### [0041]

When next, writing , reading for FIFO device comes, you explain concerning the operation of storage device subsystem 3.

#### [0042]

First, referring to flowchart of Figure 8 , when doing writing from host 1 vis-a-vis FIFO device , you explain treatment.

Before starting writing , reading for FIFO device , initialization of FIFO device is done.

With initialization , state which is not written on disk drive 33 at all with flag 41 of table 40 of Figure 7 as all 0, is made.

In addition, read address table 42 record device address , initial value makes 0.

ライトアドレステーブル 43 は CCHHR を記録するものとし、初期値は(CC,HH,R)=(0,0,1)とする。

初期化は、設定用コンソール 36 から FIFO デバイスに対して初期化の命令を出すか、ホストから初期化用のコマンドを記憶装置サブシステム 3 に出すことで、制御装置 35 によって行われる。

#### 【0043】

ホスト 1 から書き込み要求が来ると、CKD インタフェース 1031 で C 部を参照して、CCHHR アドレスを抽出して制御装置 35 に送る(ステップ 1101)。

次に、制御装置 35 では、送られてきた CCHHR から、実際のディスク 33 上の書き込み位置 CCHHR を算出する(ステップ 1102)。

シリンド番号の算出は、(ホストから指定されたシリンド番号) mod (FIFO デバイスのシリンド数)から算出できる。

FIFO デバイスのシリンド数とは、実際のディスク 33 上にできる図 6B のイメージでの CKD シリンドの数のことである。

ヘッド番号およびレコード番号は、上記のような変換の必要はない。

これにより、ディスク 33 内の最終シリンド、最終トラックまで書き込んだ場合、次の書き込み位置は、シリンド 0、すなわちディスク 33 の先頭になる。

次に、ライトアドレステーブル 43 の内容と算出した CCHHR とを比較し、異なる場合にはホストにエラーを返す(ステップ 1103)。

#### 【0044】

次に、テーブル 40 内の今回書き込み対象の CCHHR(ステップ 1102 で算出したもの)に対応するフラグ 41 の内容を参照し、0 かどうかチェックする(ステップ 1104, 1105)。

0 の場合には、制御装置 35 で当該 CCHHR(ステップ 1102 で算出したもの)を LBA に変換し、図 6B のように対象 LBA にレコードを書き込む(ステップ 1106)。

なお、書き込むレコードの C 部に含まれる CCHHR はホスト 1 から与えられた CCHHR である。

フラグ 41 が 1 の場合には、0 になるまで待ち続ける(ステップ 1104, 1105)。

ステップ 1107 では、テーブル 40 内の書き込み対象レコードのフラグ 41 を 1 に更新する。

write address table 43 record CCHHR, initial value does (CC, HH, R ) = (0, 0 and 1) with.

initialization puts out instruction of initialization from console 36 for settingvis-a-vis FIFO device , or by fact that command for initialization isput out to storage device subsystem 3 from host , is done with control device 35.

#### [0043]

When writing request comes from host 1, referring to C part with CKDinterface 1031, extracting CCHHRaddress , you send to control device 35, (step 1101 ).

Next, with control device 35, from CCHHR which is sent, writing location CCHHR on the actual disc 33 is calculated (step 1102 ).

It can calculate calculation of cylinder number , from (cylinder number which is appointed from host ) mod (Quantity of cylinder of FIFO device ).

Quantity of cylinder of FIFO device , they are quantity of CKDcylinder with image of Figure 6B which can be made on actual disc 33.

head number or record number as description above do not have necessityof conversion.

Because of this, when you wrote to final cylinder , final track inside disc 33, thefollowing writing location becomes head of cylinder 0, namely disc 33.

Next, content of write address table 43 CCHHR which was calculated iscompared, when it differs, error is returned to host , (step 1103 ).

#### [0044]

Next, you refer to content of flag 41 which this time inside the table 40 corresponds to CCHHR (Those which were calculated with step 1102. ) of writing object , whether or not 0 the check you do, (step 1104, 1105) .

In case of 0, this said CCHHR (Those which were calculated with step 1102. ) is converted to LBA with control device 35,like Figure 6B record is written to object LBA, (step 1106) .

Furthermore, as for CCHHR which is included in C part of the record which is written it is a CCHHR which is given from host 1.

When flag 41 1 is, until it becomes 0, it continues to wait, (step 1104, 1105) .

With step 1107, flag 41 of writing object record inside table 40 is renewed to1.

## 【0045】

次に、ステップ 1108 では、今回書き込んだレコードのレコード番号 R(ステップ 1102)で算出したものを見て、トラック内の最大レコード数と等しいかチェックする。

FIFO デバイスでは固定長で 1 種類のレコード長のみを扱うため、トラック内の最大レコード数は常に一定である。

等しくない場合は、ライトアドレステーブル 43 のレコード番号 R に 1 加算し(ステップ 1114)、処理を終える。

等しい場合には、ライトアドレステーブル 43 の CCHHR を R=1 とし、さらに HH の内容に 1 加算する(ステップ 1109)。

さらに、HH が 1 シリンダ内最大トラック数以上になった場合には(ステップ 1110)、CC の内容に 1 加算し、HH=0 とする(ステップ 1111)。

さらに、CC がディスク 33 の最大シリンダ数以上になった場合には(ステップ 1112)、CC=0 にする(ステップ 1113)。

## 【0046】

次に、図 9 のフローチャートを参照して、ホスト 2 から FIFO デバイスの読み出しを行ったときの処理を説明する。

最初にホスト 2 からブロックの読み出し要求が来ると、ホスト 2 から来た LBA から実際に FIFO デバイスのデバイスアドレスを算出する(ステップ 1201)。

デバイスアドレスの算出は、(ホストから来たブロック番号) mod (FIFO デバイスのブロック数)から算出できる。

FIFO デバイスのブロック数とは、図 6C のイメージでディスク 33 上に実際に作成できるデバイスアドレス単位のアドレスブロックの最大値を指す。

## 【0047】

次に、算出したデバイスアドレスをリードアドレステーブル 42 の値と比較し、異なっている場合にはエラーにする(ステップ 1202)。

等しい場合は、算出したデバイスアドレスからリード対象の CCHHR を算出する(ステップ 1203)。

デバイスアドレスから CCHHR を算出する式について既に説明した。

次に、テーブル 40 内のリード対象 CCHHR のフラグ 41 が、1 かどうかチェックする(ステップ

## [0045]

Next, with step 1108, looking at record number R (Those which were calculated with step 1102) of record which thistime was written, it is equal to quantity of maximum record inside the track , or check does.

In order with FIFO device to handle only record length of 1 kind with fixed length , quantity of maximum record inside track is normally fixed.

Case it is not equal, 1 it adds to record number R of write address table 43 and (step 1114 ), itfinishes treatment.

When it is equal, it designates CCHHR of write address table 43 as R=1,furthermore 1 adds to content of HH (step 1109 ).

Furthermore, when HH is above quantity of maximum track inside1 cylinder , (step 1110 ), 1 it adds to content of CC, makes HH=0 (step 1111 ).

Furthermore, when CC is above quantity of maximum cylinder of the disc 33, (step 1112 ), it makes CC=0, (step 1113 ).

## [0046]

Next, referring to flowchart of Figure 9 , when doing reading of the FIFO device from host 2, you explain treatment.

When reading demand for block comes from host 2 first, device address of FIFO device is calculated actually from LBA which comes from host 2 (step 1201 ).

It can calculate calculation of device address , from (block number which comes from host ) mod (Quantity of block of FIFO device ).

Quantity of block of FIFO device , it points to maximum value of the address block of device address unit which with image of Figure 6 C can be drawn upactually on disc 33.

## [0047]

When next, device address which was calculated by comparison with valueof read address table 42, it differs, it makes error , (step 1202 ).

When it is equal, CCHHR of lead object is calculated from device address which was calculated (step 1203 ).

Concerning formula which calculates CCHHR from device address already you explained.

Whether or not next, flag 41 of lead object CCHHR inside table 40, 1 the check it does, (step 1204, 1205 ).

1204、1205)。

1 の場合には次のステップに進めるが、0 の場合には1 になるまで待ち続ける。

さらに、対象 CCHHR のデータを読み出し(ステップ 1206)、その CCHHR のデータに含まれるブロックデータがすべて読み出されたなら、その CCHHR に対応するフラグ 42 を0に変更する(ステップ 1207)。

次に、もし CKD インタフェース 1031 からの書き込みレコード長が固定長ブロック1 ブロック分(デバイスアドレス 1 つ分)であればリードアドレステーブル 42 の値に 1 加算し、複数ブロック分であればそのブロック数の分だけ加算する(ステップ 1208)。

#### 【0048】

また、もしリードアドレステーブル 42 の値が FIFO デバイスの最大ブロック数以上になった場合には、リードアドレステーブル 42 の内容を 0 に戻す(ステップ 1208、1209)。

これにより、FIFO デバイス内の最終ブロックまで書き込んだ場合、次の書き込みブロック番号は 0、すなわち FIFO デバイスの先頭になる。

#### 【0049】

上記図 8 および図 9 では、ホスト 1 からデータを書き込み、ホスト 2 で読み出してデータ転送を行う例を説明したが、次に、ホスト 2 から書き込み、ホスト 1 で読み出しを行うときの処理について説明する。

#### 【0050】

まず図 10 のフローチャートを参照して、ホスト 2 から FIFO デバイスに対して書き込みを行ったときの処理を説明する。

書き込み前に FIFO デバイスの初期化を行うことは先の例と同じである。

ホスト 2 から書き込み要求が来ると、ホスト 2 から送られて来た LBA(ディスク 33 を実際の容量よりも大きな容量のディスクとして見た場合のブロック番号)から FIFO デバイスのデバイスアドレス(ディスク 33 上の実際のブロック番号)を算出する(ステップ 1301)。

この算出は、

(ホストから来たブロック番号) mod (ディスク 33 内のブロック数)

から算出できる。

table 40, 1 the check it does, (step 1204, 1205 ).

In case of 1 it advances to following step , but in case of 0 until it becomes 1, it continues to wait.

Furthermore, if reading (step 1206 ), block data which is included in data of CCHHR read out data of object CCHHR entirely, flag 42 which corresponds to CCHHR is modified in 0 (step 1207 ).

If if next, writing record length from CKDinterface 1031 is fixed length block 1block amount (device address one amount), 1 it adds to value of read address table 42, if it is a plural block amount, equal to amounts of quantity of block add (step 1208 ).

#### [0048]

In addition, if value of read address table 42 is above quantity of maximum block of FIFO device , content of read address table 42 is reset to 0, (step 1208, 1209).

Because of this, when you wrote to final block inside FIFO device , the following writing block number 0, namely becomes head of FIFO device .

#### [0049]

With above-mentioned Figure 8 and Figure 9 , from host 1 data reading \*example which does data transfer was explained with writing , host 2, but when next, from host 2 doing reading with writing , host 1, you explain concerning treatment.

#### [0050]

First referring to flowchart of Figure 10 , when doing writing from host 2 vis-a-vis FIFO device , you explain treatment.

It is same as example ahead to do initialization of FIFO device before writing .

When writing request comes from host 2, device address (actual block number on disc 33 ) of FIFO device is calculated from LBA (block number when you saw disc 33 in comparison with actual capacity as the disc of large capacity ) which is sent from host 2 (step 1301 ).

As for this calculation,

(block number which comes from host ) mod (Quantity of block inside disc 33 )

Empty it can calculate.

## 【0051】

次に、算出したデバイスアドレスをライトアドレステーブル 43 の値と比較し、異なっている場合はエラーにする(ステップ 1302)。

等しい場合は、算出したデバイスアドレスからライト対象位置 CCHHR を算出する(ステップ 1303)。

次に、テーブル 40 内のライト対象位置 CCHHR のフラグ 41 が、0 かどうかチェックする(ステップ 1304、1305)。

0 の場合には次のステップに進めるが、1 の場合には 0 になるまで待ち続ける。

ステップ 1306 では書き込み内容を FIFO デバイスのライト対象 CCHHR の位置に書き込む。

書き込み時には制御装置 35 で、デバイスアドレスをディスク 33 の LBA に変換し、また書き込み内容も図 6B の形式にして書き込む。

なお、書き込み内容の C 部に含まれる CCHHR は、ホスト 2 から与えられたデバイスアドレスを上述した式で CCHHR に変換したものである。

## 【0052】

ステップ 1307 では、テーブル 40 内の今回書き込んだ CCHHR に対応するフラグ 42 を 1 にする。

ステップ 1308 でライトアドレステーブル 43 を更新する。

このステップ 1308 も、図 8 の例と同様に、CKD インタフェース 1031 からの書き込みレコード長が固定長ブロックの何ブロック分にあたるかによって、加算する値が決まる。

もし、ライトアドレステーブル 43 の値が FIFO デバイスの最大ブロック数以上になった場合には、ライトアドレステーブル 43 の内容を 0 に戻す(ステップ 1309、1310)。

これにより、FIFO デバイス内の最終ブロックまで書き込んだ場合、次の書き込みブロック番号は 0、すなわち FIFO デバイスの先頭になる。

## 【0053】

図 11 のフローチャートを参照して、ホスト 1 から FIFO デバイスの読み出しを行ったときの処理を説明する。

ホスト 1 からのブロックの読み出し要求が来ると、CKD インタフェース 1031 でホスト 1 から与えられた C 部を参照して、CCHHR アドレスを抽出

## [0051]

When next, device address which was calculated by comparison with value of write address table 43, it differs, it makes error, (step 1302).

When it is equal, write target position CCHHR is calculated from device address which was calculated (step 1303).

Whether or not next, flag 41 of write target position CCHHR inside table 40, 0 the check it does, (step 1304, 1305).

In case of 0 it advances to following step, but in case of 1 until it becomes 0, it continues to wait.

With step 1306 writing content is written to location of write object CCHHR of the FIFO device.

With control device 35, it converts device address to LBA of disc 33 at the time of writing, it writes in addition writing content in form of the Figure 6B.

Furthermore, as for CCHHR which is included in C part of the writing content, it is something which is converted to CCHHR with the formula which device address which is given from host 2 description above is done.

## [0052]

With step 1307, flag 42 which corresponds to CCHHR which this time inside table 40 was written is designated as 1.

write address table 43 is renewed with step 1308.

this step 1308, in same way as example of Figure 8, writing record length from CKD interface 1031 hits to no block amount of fixed length block, value which is added is decided.

When value of write address table 43 is above quantity of maximum block of the FIFO device, content of write address table 43 is reset to 0, (step 1309, 1310).

Because of this, when you wrote to final block inside FIFO device, the following writing block number 0, namely becomes head of FIFO device.

## [0053]

Referring to flowchart of Figure 11, when doing reading of FIFO device from host 1, you explain treatment.

When reading demand for block from host 1 comes, referring to the C part which with CKD interface 1031 is given from host 1, extracting the CCHHR address, you send to control

して制御装置 35 に送る(ステップ 1401)。

次に、制御装置 35 では、送られてきた CCHHR から実際のディスク 33 の読み出し位置 CCHHR を算出する(ステップ 1402)。

シリンド番号の算出は、(ホストから指定されたシリンド番号) mod (FIFO デバイスのシリンド数)から算出できる。

FIFO デバイスのシリンド数は、図 8 のステップ 1102 で説明したのと同じである。

ヘッド番号およびレコード番号は、上記のような変換をせず、そのまま使用する。

これにより、ディスク 33 内の最終シリンド、最終トラックまで読み出した場合、次の読み出し位置はシリンド 0、すなわちディスク 33 の先頭になる。

次に、リードアドレステーブル 42 の内容と算出した CCHHR を比較し、異なる場合にはホストにエラーを返す(ステップ 1403)。

#### 【0054】

次に、テーブル 40 内の読み出し対象の CCHHR に対応するフラグ 41 が、1 かどうかチェックする(ステップ 1404、1405)。

1 の場合には次のステップに進めるが、0 の場合には 1 になるまで待ち続ける。

ステップ 1406 で読み出し対象の CCHHR のレコードを読み出し、ステップ 1407 で当該 CCHHR に対応するフラグ 41 を 0 に更新する。

#### 【0055】

次に、ステップ 1408 では、今回読み出した位置である CCHHR のレコード番号 R を見て、トラック内の最大レコード数と等しいかチェックする。

FIFO デバイスでは固定長で 1 種類のレコード長のみを扱うため、トラック内の最大レコード数は常に一定である。

等しくない場合は、ライトアドレステーブル 43 のレコード番号 R に 1 加算し(ステップ 1414)、処理を終える。

等しい場合には、ライトアドレステーブル 43 の CCHHR を R=1 とし、さらに HH の内容に 1 加算する(ステップ 1409)。

さらに、HH が 1 シリンド内最大トラック数以上になつた場合には、CC の内容に 1 加算し、HH=0 にする(ステップ 1410、1411)。

さらに CC がディスク 33 の最大シリンド数を越え

device 35, (step 1401)。

Next, with control device 35, read position CCHHR of actual disc 33 is calculated from the CCHHR which is sent (step 1402).

It can calculate calculation of cylinder number , from (cylinder number which is appointed from host ) mod (Quantity of cylinder of FIFO device ).

Quantity of cylinder of FIFO device , that you explained with step 1102 of Figure 8 , is same.

head number or record number do not do, use as description above conversionthat way.

Because of this, reading it is to final cylinder , final track inside disc 33 when, following read position becomes head of cylinder 0, namely disc 33.

Next, content of read address table 42 CCHHR which was calculated is compared, when it differs, error is returned to host , (step 1403) .

#### [0054]

Whether or not next, flag 41 which corresponds to CCHHR of the reading object inside table 40, 1 check it does, (step 1404, 1405) .

In case of 1 it advances to following step , but in case of 0 until it becomes 1, it continues to wait.

record of CCHHR of reading object flag 41 which corresponds to the this said CCHHR with reading , step 1407 is renewed to 0 with step 1406.

#### [0055]

Next, with step 1408, this time reading it is looking at record number R of CCHHR which is a location , it is equal to quantity of maximum record inside track , or check does.

In order with FIFO device to handle only record length of 1 kind with fixed length , quantity of maximum record inside track is normally fixed.

Case it is not equal, 1 it adds to record number R of write address table 43 and (step 1414) , it finishes treatment.

When it is equal, it designates CCHHR of write address table 43 as R=1,furthermore 1 adds to content of HH (step 1409) .

Furthermore, when HH is above quantity of maximum track inside 1 cylinder , 1 it adds to content of CC, makes HH=0 (step 1410, 1411) .

Furthermore when CC exceeds quantity of maximum cylinder

た場合には CC=0 にする(ステップ 1412、1413)。

#### 【0056】

図 8 および図 9 の動作を並行して実施、または図 10 および図 11 の動作を並行して実施することで、記憶装置サブシステム 3 は FIFO として動作する。

ホスト 1 とホスト 2 のデータ形式が互いに異なるため、データ形式の変換が必要であるが、この変換も記憶装置サブシステム内で行われる。

#### 【0057】

なお、ホスト 1 または 2 から FIFO デバイスにデータ書き込みを行う際に該書き込みデータの文字コードを変換して書き込むようにしたり、ホスト 1 または 2 で FIFO デバイスからデータ読み出しを行う際に該読み出しデータの文字コードを変換して読み出すようにしてもよい。

そのためには、例えば、図 3 のステップ 104、図 4 のステップ 204、図 8 のステップ 1106、図 9 のステップ 1206、図 10 のステップ 1306、または図 11 のステップ 1406 などで書き込みまたは読み出しを行う際にコード変換を行うようにすればよい。

これにより、ホスト 1 で用いているコードが EBCDIC で、ホスト 2 で用いているコードが ASCII であるような場合でも、相互にコード変換を行うことができる。

#### 【0058】

##### 【発明の効果】

以上説明したように、本発明に係る記憶装置サブシステムによれば、記憶手段を先入れ先出し機構(FIFO)として使用することができ、ホスト間の高速なデータ転送に利用できる。

さらに、記憶装置内にデータ形式の変換機構を設けることで、異なったデータ形式の計算機間でのデータ転送も、計算機側で特別な変換作業なしに行うことができる。

##### 【図面の簡単な説明】

##### 【図1】

本発明を適用した記憶装置サブシステムの構成例を示す図

##### 【図2】

記憶装置サブシステム内制御メモリで保有するテーブルとアドレステーブルの例を示す図

of disc 33, it makes CC=0, (step 1412, 1413).

#### [0056]

Operation of Figure 8 and Figure 9 in parallel is executed, or by fact that inparallel, it executes operation of Figure 10 and Figure 11 , storage device subsystem 3 operatesas FIFO .

data form of host 1 and host 2 mutually because of different ,conversion of data form is necessary, but also this conversion isdone inside storage device subsystem .

#### [0057]

Furthermore, converting character code of said writing data occasion where from host 1 or 2 data writing is done in FIFO device , converting character code of the said reading data occasion which it tries to write, with host 1 or 2 from the FIFO device data reading does, it is possible to read out.

For that, it should have tried to convert code occasion where the writing or reading is done with such as step 1306, of step 1206, Figure 10 of the step 1106, Figure 9 of step 204, Figure 8 of step 104, Figure 4 of for example Figure 3 or step 1406 of Figure 11 .

Because of this, code which has been used with host 1 being EBCDIC „, it is possible even with kind of when code which hasbeen used with host 2 is ASCII to convert mutually code .

#### [0058]

##### [Effects of the Invention]

As above explained, according to storage device subsystem which relates to this invention ,it can use storage means in high speed data transfer between host can utilize the first-in, first-out mechanism (FIFO ) as.

Furthermore, by fact that conversion mechanism of data form is providedinside storage device , data transfer between computer of data form whichdiffers, it is possible on computer side to do in special conversionjob none .

##### [Brief Explanation of the Drawing(s)]

##### [Figure 1]

Figure which shows configuration example of storage device subsystem which applies this invention

##### [Figure 2]

Figure which shows example of table and address table whichare possessed with control memory inside storage device subsystem

## 【図3】

FIFO デバイスに対するライト処理の流れを示す図

## [Figure 3]

Figure which shows flow of write treatment for FIFO device

## 【図4】

FIFO デバイスに対するリード処理の流れを示す図

## [Figure 4]

Figure which shows flow of lead treatment for FIFO device

## 【図5】

本発明を適用した記憶装置サブシステムの構成例(第2の実施形態)を示す図

## [Figure 5]

Figure which shows configuration example (second embodiment ) of storage device subsystem which applies the this invention

## 【図6】

記憶装置サブシステム内のディスクに CKD 形式で格納される様子、およびホストコンピュータから見えるデータ形式を示す図

## [Figure 6]

Figure which shows data form which is visible from circumstances, and host computer which in disc inside storage device subsystem are housed with the CKDform

## 【図7】

記憶装置サブシステム内制御メモリで保有するテーブルとアドレステーブルの例を示す図

## [Figure 7]

Figure which shows example of table and address table which are possessed with control memory inside storage device subsystem

## 【図8】

FIFO デバイスに対するライト処理の流れを示す図

## [Figure 8]

Figure which shows flow of write treatment for FIFO device

## 【図9】

FIFO デバイスに対するリード処理の流れを示す図

## [Figure 9]

Figure which shows flow of lead treatment for FIFO device

## 【図10】

FIFO デバイスに対するライト処理の流れを示す図

## [Figure 10]

Figure which shows flow of write treatment for FIFO device

## 【図11】

FIFO デバイスに対するリード処理の流れを示す図

## [Figure 11]

Figure which shows flow of lead treatment for FIFO device

## 【符号の説明】

## [Explanation of Symbols in Drawings]

1

1

ホスト

host

1031

1031

CKD インタフェース

CKDInterface

1032

1032

固定長インターフェース

fixed length interface

2

2

ホスト

host

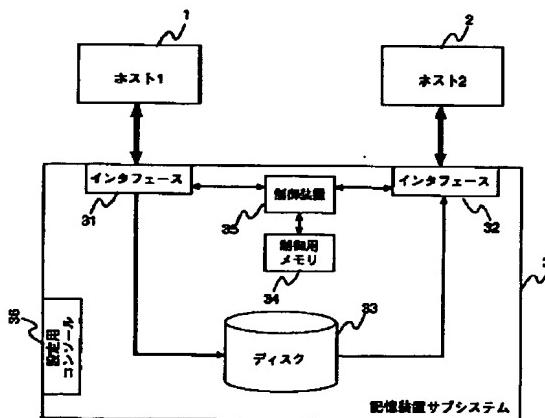
3	3
記憶装置サブシステム	storage device subsystem
31	31
インターフェース	interface
32	32
インターフェース	interface
33	33
ディスク	disc
34	34
制御用メモリ	memory for control
35	35
制御装置	control device
36	36
設定用コンソール	console for setting
40	40
テーブル	table
41	41
フラグ	flag
42	42
リードアドレステーブル	read address table
43	43
ライトアドレステーブル	write address table
51	51
HA	HA
52	52
R0C	R0C
53	53
R0D	R0D
54	54
R1C	R1C
55	55
R1D	R1D
56	56
R2C	R2C
57	57

R2D	R2D
60	60
LBA0	LBA0
61	61
LBA1	LBA1
62	62
LBA2	LBA2
63	63
LBA3	LBA3
64	64
LBA4	LBA4
65	65
LBA5	LBA5
71	71
デバイスアドレス	device address

**Drawings**

【図1】

[Figure 1]

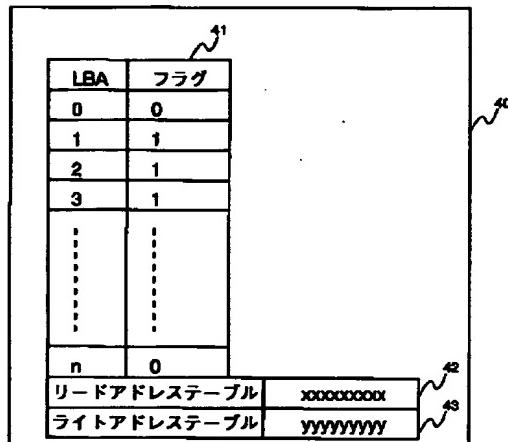


【図2】

[Figure 2]

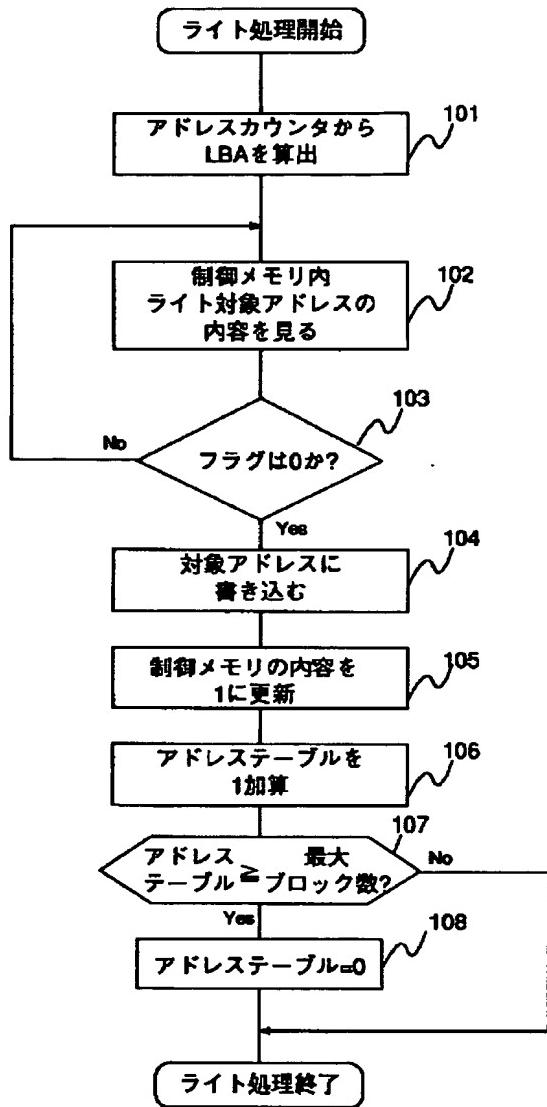
JP2000163246A

2000-6-16



【図3】

[Figure 3]



【図7】

[Figure 7]

JP2000163246A

2000-6-16

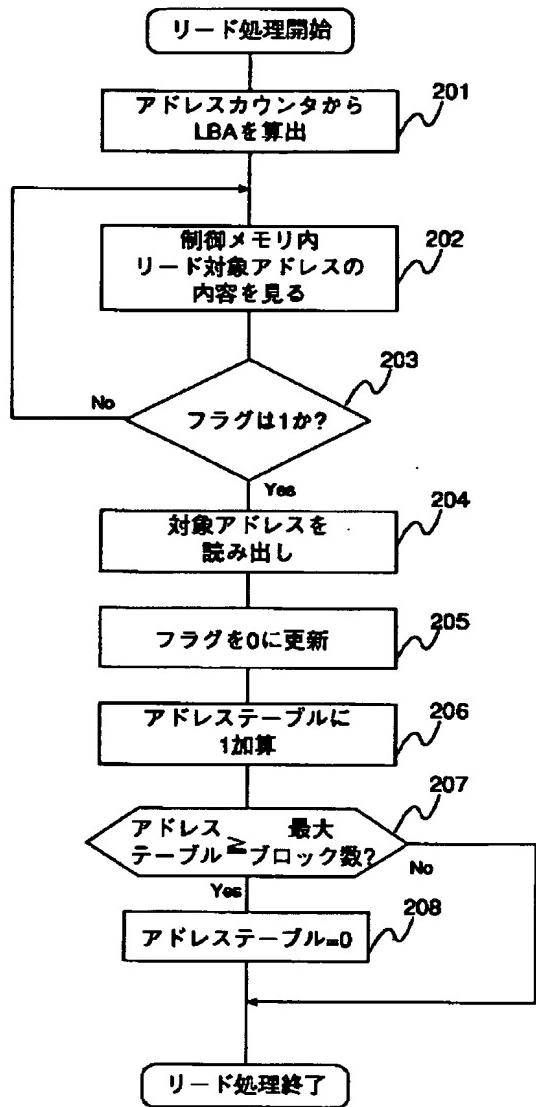
The diagram illustrates a memory structure. At the top left is a table with four rows and two columns, labeled 'CCHHR' and 'フラグ'. The first three rows have values '0,0,1', '0,0,2', and '0,1,1' respectively in the first column, and '0', '1', and '1' in the second column. The fourth row has empty cells for both columns. To the right of this table is a large empty rectangular area. Below the table is another table with four rows and two columns. The first row has 'n,m,2' in the first column and '0' in the second. The second row has 'リードアドレステーブル' in the first column and 'xxxxxxxxxx' in the second. The third row has 'ライトアドレステーブル' in the first column and 'yyyyyyyyyy' in the second. The fourth row is empty. Wavy lines labeled 41 through 43 point to the top edge of the first table, the right edge of the large empty area, and the right edge of the second table, respectively.

CCHHR	フラグ
0,0,1	0
0,0,2	1
0,1,1	1
0,1,2	1
n,m,2	0

リードアドレステーブル	xxxxxxxxxx
ライトアドレステーブル	yyyyyyyyyy

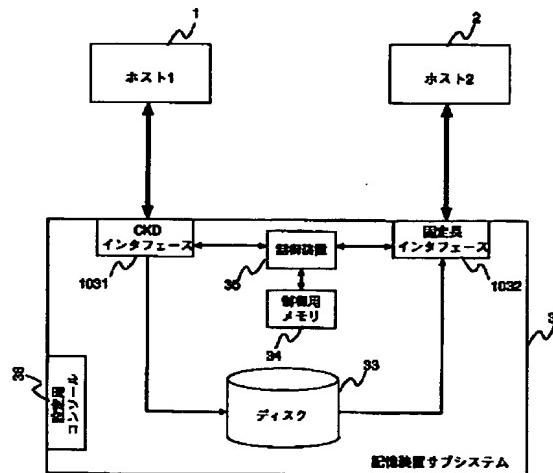
【図4】

[Figure 4]



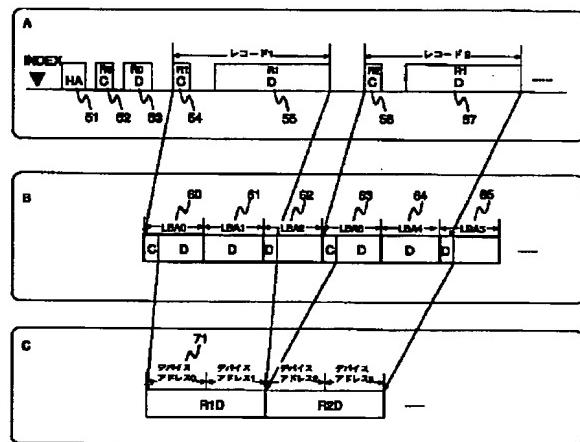
【図5】

[Figure 5]



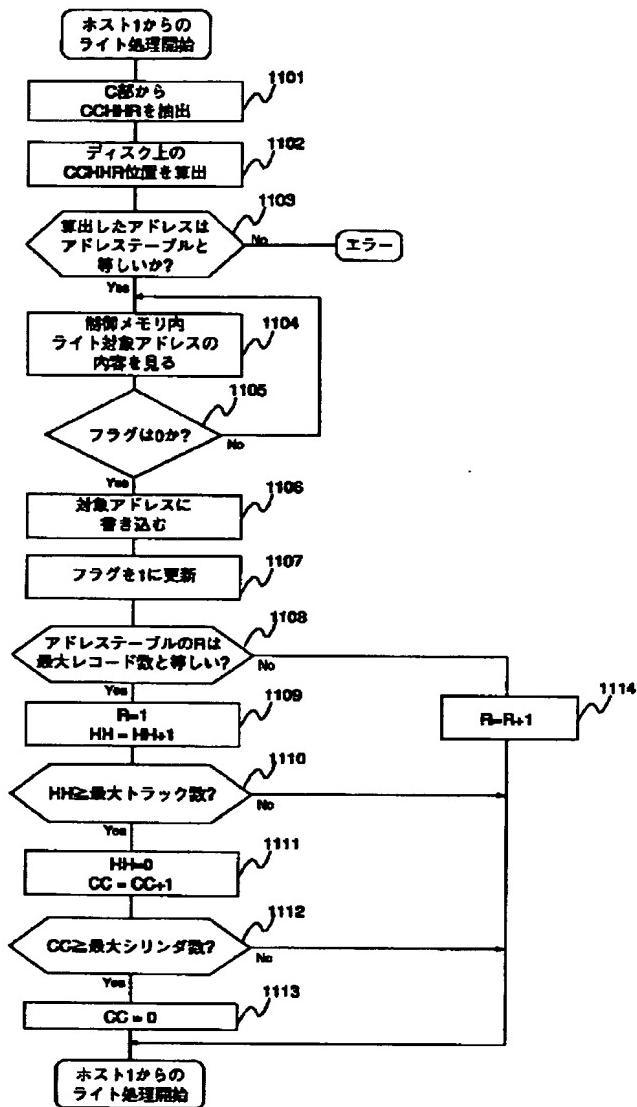
【図6】

[Figure 6]



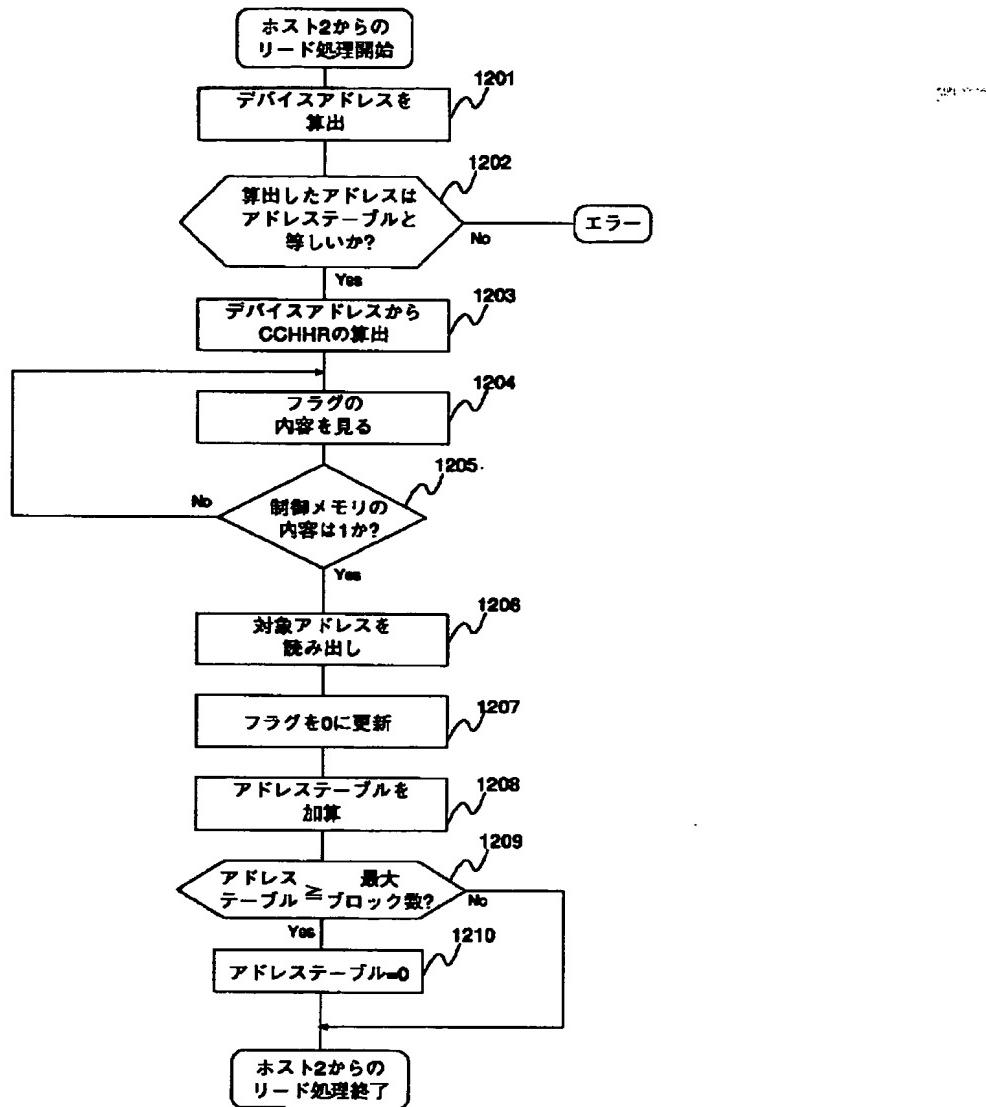
【図8】

[Figure 8]



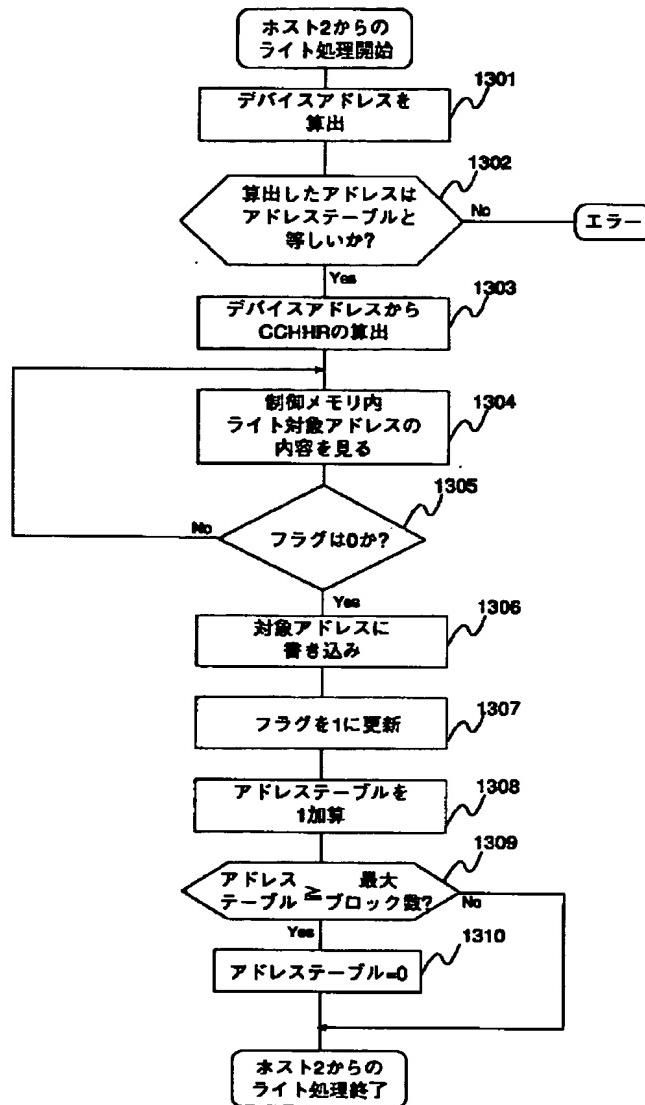
【図9】

[Figure 9]



【図10】

[Figure 10]



【図11】

[Figure 11]

